



Programmation Réactive sur Android avec RxJava2

Durée 1 jour(s) (RX-ANDROID)

**Approfondissez l'utilisation de RxJava2 pour des applications Android
plus performantes et stables**

Description

Sur le temps d'une journée complétez votre formation sur la programmation réactive avec RxJava. Vous réaliserez une application Android réactive de bout en bout, type client Instagram. Vous mettrez en oeuvre les notions les plus récurrentes : - Client Http réactif avec Retrofit2 + OkHttp3 - Agrégation d'API complexes - Moteur de recherche efficient - Mise en cache des appels Http

Formation animée en présentiel

La formation en présentiel se déroule sur des jours consécutifs

Formation disponible en mode "formation à distance"

La formation à distance peut se dérouler sur des jours consécutifs ou se décomposer en demies journées

Objectifs

- Savoir intégrer RxJava2 dans une application Android
- Maîtriser la librairie RxJava2 (Développement et Test)
- Savoir intégrer de bout en bout une architecture réactive (UI, Persistance, Service, Client Http)

Public

- Développeur Android

Prérequis

- Connaissance en programmation réactive
- Connaissance du langage Kotlin
- Connaissance du framework Android et de la librairie Retrofit

Répartition

0% Théorie, 100% Pratique

Programme

Les basiques

- Déclaration d'un Client Http avec Retrofit2 & OkHttp3
- Réalisation d'une première requête Http
- Maîtriser le mécanisme de souscription spécifique à Android
- Afficher la réponse sur un écran
- Lier les traitements asynchrones avec le cycle de vie d'un Fragment

Aggregation d'API Web complexes

- Concevoir un flux complexe effectuant de multiples appels Http
- Utilisation d'opérateurs de type flatMap & compose
- Utilisation de l'API Transformer afin de réutiliser des parties de flux

Implémenter une recherche efficiente

- Appliquer des filtres à votre recherche
- Optimisation de la recherche
- Gestion d'un grand nombre de résultats

Implémenter un cache

- Implémenter un cache in memory
- Gérer la négociation entre le cache et les appels http